DevOps Shack

TRIVY Interview Questions

[Click Here To Enrol To Batch-5 | DevOps & Cloud DevOps]

1. **What is Trivy?**
    - **Answer:** Trivy is an open-source vulnerability scanner for containers and other artifacts, such as filesystems and Git repositories. It identifies vulnerabilities by scanning images and reports on any known vulnerabilities, configuration issues, and misconfigurations.
2. **How does Trivy work?**
    - **Answer:** Trivy works by downloading a vulnerability database and scanning container images or filesystems against this database. It identifies vulnerabilities by comparing the package versions found in the image or filesystem with known vulnerabilities in its database.
3. **What types of vulnerabilities does Trivy detect?**
    - **Answer:** Trivy detects vulnerabilities in operating system packages (e.g., Alpine, RHEL, CentOS), application dependencies (e.g., RubyGems, npm, Python), and configuration issues in Kubernetes manifests.
4. **What are the primary use cases for Trivy?**
    - **Answer:** Trivy is primarily used for scanning container images for vulnerabilities before deployment, integrating vulnerability scanning into CI/CD pipelines, and ensuring compliance with security standards.
5. **What platforms does Trivy support?**
    - **Answer:** Trivy supports multiple platforms, including Linux and macOS. It can be used on Windows through WSL (Windows Subsystem for Linux).

Installation and Setup

6. **How do you install Trivy?**
    - **Answer:** Trivy can be installed using package managers like `brew` for macOS or `apt` for Debian-based systems. It can also be installed via Docker:

      ```
      brew install aquasecurity/trivy/trivy
      ```

      or

      ```
      sudo apt-get install -y trivy
      ```

      or

      ```
      docker pull aquasec/trivy
      ```

7. **How do you update Trivy to the latest version?**
    - **Answer:** To update Trivy, you can use the package manager's update commands:

      ```
      brew upgrade trivy
      ```

      or

      ```
      sudo apt-get update && sudo apt-get install --only-upgrade trivy
      ```

      or pull the latest Docker image:

      ```
      docker pull aquasec/trivy:latest
      ```

8. **How do you configure Trivy to use a specific vulnerability database?**
    - **Answer:** Trivy uses the default vulnerability database from Aqua Security. You can configure a custom database by setting the `TRIVY_DB_REPOSITORY` environment variable to the desired database URL.

9. **What is the command to initialize Trivy's vulnerability database?**
    - **Answer:** Trivy automatically downloads the database on its first run. However, you can manually initialize it using:

      ```
      trivy --download-db-only
      ```

10. **How do you use Trivy in an offline environment?**
    - **Answer:** To use Trivy offline, download the database on a machine with internet access:

      ```
      trivy --download-db-only
      ```

Copy the downloaded database (`~/.cache/trivy/db`) to the offline environment and set the `TRIVY_CACHE_DIR` environment variable to the database location.

Scanning Container Images

11. **How do you scan a Docker image using Trivy?**
    - **Answer:** Use the following command to scan a Docker image:

      ```
      trivy image <image_name>
      ```

12. **How do you scan a local Docker image with Trivy?**
    - **Answer:** Use the image name or ID for a local image:

      ```
      trivy image <local_image_name_or_id>
      ```

13. **How do you scan a Docker image from a private registry with Trivy?**
    - **Answer:** First, log in to the private registry using Docker:

      ```
      docker login <registry_url>
      ```

      Then scan the image:

      ```
      trivy image <registry_url>/<image_name>
      ```

14. **What are the options for output formats when scanning with Trivy?**
    - **Answer:** Trivy supports various output formats including table (default), JSON, and template. Use the `--format` flag to specify the format:

      ```
      trivy image --format json <image_name>
      ```

15. **How do you save Trivy scan results to a file?**
    - **Answer:** Redirect the output to a file:

      ```
      trivy image <image_name> --format json > results.json
      ```

Vulnerability Database and Updates

16. **How frequently is Trivy's vulnerability database updated?**
    - **Answer:** Trivy's vulnerability database is updated multiple times a day to ensure it includes the latest known vulnerabilities.

17. **How can you force Trivy to update its vulnerability database?**

- **Answer:** Use the `--refresh` flag to update the database:

  ```
  trivy --refresh
  ```

18. **What is the default location of Trivy's vulnerability database?**
    - **Answer:** The default location is `~/.cache/trivy/db`.
19. **How do you configure a different cache directory for Trivy?**
    - **Answer:** Set the `TRIVY_CACHE_DIR` environment variable to the desired directory:

      ```
      export TRIVY_CACHE_DIR=/path/to/cache
      ```

20. **How do you ensure Trivy uses the latest vulnerability database in a CI/CD pipeline?**
    - **Answer:** Add a step to your pipeline to refresh the database before running scans:
    - ```
      trivy --refresh
      trivy image <image_name>
      ```

Scanning Filesystems and Repositories

21. **How do you scan a local filesystem with Trivy?**
    - **Answer:** Use the `fs` command to scan a directory:

      ```
      trivy fs /path/to/directory
      ```

22. **How do you scan a Git repository with Trivy?**
    - **Answer:** Use the `repo` command to scan a repository URL:

      ```
      trivy repo <repository_url>
      ```

23. **How do you scan a specific branch of a Git repository with Trivy?**
    - **Answer:** Use the `--branch` flag:

      ```
      trivy repo --branch <branch_name> <repository_url>
      ```

24. **How do you exclude certain directories or files from a Trivy scan?**
    - **Answer:** Use the `--ignore` flag to specify patterns to exclude:

      ```
      trivy fs --ignore /path/to/exclude /path/to/directory
      ```

25. **Can Trivy scan files other than container images and repositories?**
      ```

- **Answer:** Yes, Trivy can scan filesystems and directories, not just container images and repositories.

Severity Levels and Reporting

26. **What are the different severity levels in Trivy reports?**
    - **Answer:** Severity levels include UNKNOWN, LOW, MEDIUM, HIGH, and CRITICAL.
27. **How do you filter Trivy scan results by severity?**
    - **Answer:** Use the `--severity` flag to specify the severity levels to include:

      ```
      trivy image --severity HIGH,CRITICAL <image_name>
      ```
28. **How do you only show vulnerabilities introduced by the application code in Trivy?**
    - **Answer:** Use the `--vuln-type` flag to filter for application-specific vulnerabilities:

      ```
      trivy image --vuln-type library <image_name>
      ```
29. **How do you include all vulnerability types in a Trivy scan?**
    - **Answer:** Use the `--vuln-type` flag with `os,library` to include both OS and library vulnerabilities:

      ```
      trivy image --vuln-type os,library <image_name>
      ```
30. **How do you generate a report with Trivy in a specific format?**
    - **Answer:** Use the `--format` flag to specify the format (e.g., JSON, table):

      ```
      trivy image --format json <image_name>
      ```

Integration with CI/CD

31. **How do you integrate Trivy with a CI/CD pipeline?**
    - **Answer:** Add a Trivy scan step in your CI/CD pipeline script. For example, in a Jenkins pipeline:

```
stage('Scan with Trivy') {
  steps {
    sh 'trivy image <image_name>'
  }
}
```

32. **How do you fail a CI/CD pipeline if Trivy finds vulnerabilities?**

- **Answer:** Use the `--exit-code` flag to specify the exit code for vulnerabilities:

  ```
  trivy image --exit-code 1 <image_name>
  ```

33. **How do you exclude certain vulnerabilities from failing the CI/CD pipeline?**

    - **Answer:** Use the `--ignore-unfixed` flag to ignore unfixed vulnerabilities or the `--ignorefile` to specify a file with vulnerabilities to ignore.

34. **How do you scan Kubernetes manifests in a CI/CD pipeline using Trivy?**

    - **Answer:** Use the `config` command to scan manifests:

      ```
      trivy config /path/to/manifests
      ```

35. **How do you use Trivy with GitHub Actions?**

    - **Answer:** Add a Trivy scan step in your GitHub Actions workflow file:
    - `steps`

: - name: Run Trivy scan run: trivy image <image_name> ```

Advanced Usage

36. **How do you configure Trivy to use a proxy server?**

    - **Answer:** Set the `HTTP_PROXY`, `HTTPS_PROXY`, and `NO_PROXY` environment variables to configure proxy settings for Trivy.

37. **How do you scan a container image without pulling it locally using Trivy?**

    - **Answer:** Use Trivy with Docker to scan an image without pulling it:

      ```
      trivy image --remote <image_name>
      ```

38. **What are the common error codes in Trivy and their meanings?**

    - **Answer:** Common error codes include:
      - `0`: No vulnerabilities found
      - `1`: Vulnerabilities found
      - `2`: Operational errors (e.g., network issues, file not found)

39. **How do you handle rate limiting when using Trivy with Docker Hub?**

    - **Answer:** Use Docker Hub credentials to authenticate and reduce rate limiting:
    - `docker login`
      `trivy image <image_name>`

40. **How do you scan a multi-stage Dockerfile with Trivy?**

- **Answer:** Trivy scans the final image produced by the multi-stage Dockerfile. Ensure the image is built before scanning.

Troubleshooting and Best Practices

41. **What should you do if Trivy's vulnerability database fails to download?**
    - **Answer:** Check your internet connection, proxy settings, and ensure that the Trivy servers are accessible. You can also manually download the database and configure Trivy to use it.

42. **How do you reduce false positives in Trivy scans?**
    - **Answer:** Regularly update Trivy and its vulnerability database, use the latest stable version of the scanned software, and configure ignore policies for known false positives.

43. **How do you ignore certain vulnerabilities in Trivy scans?**
    - **Answer:** Use an `.trivyignore` file to list vulnerabilities to ignore:
    - `CVE-2020-12345`
    - `CVE-2019-6789`

44. **How do you scan images in a Kubernetes cluster with Trivy?**
    - **Answer:** Use Trivy Operator for Kubernetes to automatically scan images in a cluster.

45. **How do you scan a large image or repository efficiently with Trivy?**
    - **Answer:** Use a CI/CD pipeline to scan images regularly, segment scans by components, and parallelize scans if possible.

46. **What are the best practices for using Trivy in production environments?**
    - **Answer:** Integrate Trivy into CI/CD pipelines, regularly update the database, scan images before deployment, and review and address vulnerabilities promptly.

47. **How do you debug issues with Trivy scans?**
    - **Answer:** Use the `--debug` flag to enable debug logging and review the logs for detailed information on scan processes and errors:

      ```
      trivy image --debug <image_name>
      ```

48. **How do you scan images built on non-standard base images with Trivy?**
    - **Answer:** Ensure the non-standard base image is supported by Trivy or use a custom vulnerability database tailored to the base image.

49. **How do you ensure Trivy's database is up-to-date in an automated environment?**

    - **Answer:** Schedule regular updates of Trivy's database using a cron job or equivalent scheduling tool in your environment.

50. **What are the limitations of Trivy and how can they be mitigated?**

    - **Answer:** Trivy's limitations include reliance on its vulnerability database and potential rate limits from registries. Mitigate these by using up-to-date databases, authenticating with registries, and combining Trivy with other security tools for comprehensive coverage.